DOCUMENT RESUME

ED 310 114 TM 013 236

AUTHOR Adema, Jos J.
TITLE The Construction of Two-Stage Tests. Project Psychometric Aspects of Item Banking No. 39.
INSTITUTION Twente Univ., Enschede (Netherlands). Dept. of Education.
REPORT NO RR-88-14
PUB DATE Nov 88
NOTE 34p.
AVAILABLE FROM Bibliotheek, Department of Education, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.
PUB TYPE Reports - Evaluative/Feasibility (142)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS *Adaptive Testing; Computer Assisted Testing; Item Banks; *Latent Trait Theory; *Linear Programing; Mathematical Formulas; *Mathematical Models; *Test Construction
IDENTIFIERS Item Dependency Model; *Two Stage Testing

ABSTRACT

Although two-stage testing is not the most efficient form of adaptive testing, it has some advantages. In this paper, linear programming models are given for the construction of two-stage tests. In these models, practical constraints with respect to, among other things, test composition, administration time, and inter-item dependencies play an important role. Research indicates that two-stage tests can be constructed both sequentially and simultaneously. Models based on the maximin model for test construction developed by W. J. van der Linden and E. Boekkooi-Timminga (1989) are formulated for the sequential case, with constraints specified at test and subtest levels. It is assumed that a bank of items calibrated under the item response model is available and that "information" is used in accordance with G. H. Fischer's information model. The maximin design is used in order to select the items that maximize the information in the test, while the resulting test information function still has the desired shape. The paper concludes that simultaneous test construction has the disadvantage of having a large number of variables; hence, constraints must be considered. (TJH)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED310114

The construction of Two-Stage Tests

Research Report 88-14

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

J. NELISSEN

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Jos J. Adema

department of EDUCATION

Division of Educational Measurement and Data Analysis

University of Twente

TM013236

Project Psychometric Aspects of Item Banking No.39

Colofon:
Typing: Lorette Padberg
Cover design: Audiovisuele Sectie TOLAB Toegepaste Onderwijskunde
Printed by: Centrale Reproductie-afdeling

The Construction of Two-Stage Tests

Jos J. Adema

The construction of two-stage tests / Jos J. Adema -
Enschede : University of Twente, Department of Education,
November, 1988. - 24 pages

Abstract

Although two-stage testing is not the most efficient form of adaptive testing, it has some advantages. In this paper linear programming models are given for the construction of two-stage tests. In these models practical constraints with respect to, e.g., test composition, administration time, inter-item dependencies, play an important role. Two-stage tests can be constructed both sequentially and simultaneously. Models are formulated for the sequential case with constraints specified at test and subtest level. Simultaneous test construction has the disadvantage that a large number of variables and constraints have to be considered.

The construction of two-stage tests

Two-stage testing implies that the examinee is first confronted with a routing test. The choice of the second test depends on the score on the routing test.

Two-stage tests will be most valuable in situations where the group tested has a range of ability too wide to be measured effectively by a peaked conventional test (Lord, 1980, p.146). Other forms of adaptive testing are often more efficient than two-stage testing. However, two-stage tests can be useful. Their advantages are that they can be administered by paper and pencil (Lord, 1980; Fischer & Pendl, 1980), and, under the Rasch model, that the ability estimates are easily computed (Fischer & Pendl, 1980; Glas, 1988). Wainer and Kiely (1987) give a number of problems that are associated with most forms of adaptive testing, like context effects, lack of robustness, and item difficulty ordering (the items administered in the beginning are too difficult for the least able students). In general, two-stage tests need not be sensitive to these problems if they are dealt with appropriately when constructing the test.

As pointed out by Yen (1983) linear programming models (LP models) can be used for the construction of tests (although we use the word "model" an LP model is in fact a problem). LP models for the construction of two-stage tests will be given in this paper. In these models practical constraints, i.e., demands with respect to the properties of the test, are taken into account. Practical constraints can,

for instance, be used to control the test composition and the administration time. The models are based on the maximin model for test construction (van der Linden & Boekkooi-Timminga, 1989).

Further on, it is assumed that a bank of items calibrated under an item response model is available and when the word "information" is used Fisher's information is meant. In the maximin model the test constructor has to provide the relative shape of a target test information function by giving target values at certain points. The idea is to select the items such that they maximize the information in the test, while the resulting test information function still has the desired shape.

Define the decision variables x_i as follows:

$$x_i = \begin{cases} 0 & \text{item i not in the test} \\ 1 & \text{item i in the test.} \end{cases}$$

Let p_l, $l = 1,\ldots,L$ be the relative amount of information that is required at ability level θ_l and $I_i(\theta_l)$ the amount of information at ability level θ_l for item i. The maximin model can then be formulated as follows:

(1) Max. y

subject to

$$(2) \qquad \sum_{i=1}^{I} I_i(\theta_l)x_i - p_l y \geq 0, \qquad l = 1, 2, \ldots, L,$$

$$(3) \qquad \sum_{i=1}^{I} x_i = n,$$

$$(4) \qquad x_i \in \{0,1\}, \qquad i = 1, 2, \ldots, I,$$

$$(5) \qquad y \geq 0,$$

where n is the number of items in the test, I is the number of items in the item bank, and y is a dummy variable. In this representation it is clear that y can be considered a lower bound to the weighted sums of decisions variables $\Sigma_{i=1}^{I} I_i(\theta_l)\ p_l^{-1}x_i$ and that the values of x_i are selected such that this lower bound is maximal.

The maximin model is in the operations research literature known as a mixed integer linear programming model. These models can be solved by branch-and-bound methods, i.e., methods are available that maximize y and simultaneously compute the corresponding optimal values for x_i, $i = 1, \ldots, I$ (Land & Doig, 1960).

Practical Constraints

The linear programming models for the construction of two-stage tests may include a number of practical constraints as formulated by van der Linden and Boekkooi-Timminga (1989).

In most item banks we can distinguish subsets of items. Items can, for instance, be grouped in subsets on the basis of their content or format (e.g., multiple choice, completion).

Generally, three different kinds of subsets can be distinguished:

K_j ($j = 1, 2, \ldots, J_K$): At most n_{K_j} items may be selected from the subsets K_j;

E_j ($j = 1, 2, \ldots, J_E$): Exactly n_{E_j} items should be selected from the subsets E_j;

G_j ($j = 1, 2, \ldots, J_G$): At least n_{G_j} items should be selected from the subsets G_j.

The following constraints are related to the subsets given:

$$(6) \qquad \sum_{i \in K_j} x_i \leq n_{K_j}, \qquad j = 1, 2, \ldots, J_K,$$

$$(7) \qquad \sum_{i \in E_j} x_i = n_{E_j}, \qquad j = 1, 2, \ldots, J_E,$$

$$(8) \qquad \sum_{i \in G_j} x_i \geq n_{G_j}, \qquad j = 1, 2, \ldots, J_G.$$

For example, let an item bank for French be partitioned with respect to its content in a vocabulary, a grammar and a reading comprehension part and with respect to its format of items in a multiple-choice and a matching part. A test

constructor has the following demands with respect to the composition of the test:

1. The number of grammar items in the test is not greater than 5:

$$\sum_{i \epsilon K_1} x_i \leq 5,$$

where K_1 is the subset of grammar items,

2. The number of vocabulary items in the test is equal to 6:

$$\sum_{i \epsilon E_1} x_i = 6,$$

where E_1 is the subset of vocabulary items,

3. The number of reading comprehension items in the test is greater than 5:

$$\sum_{i \epsilon G_1} x_i \geq 6,$$

where G_1 is the subset of reading comprehension items,

4. The number of multiple choice items in the test is not greater than 12:

$$\sum_{i \epsilon K_2} x_i \leq 12,$$

where K_2 is the subset of multiple choice items,

5. The number of matching items in the test is greater than 6:

$$\sum_{i \epsilon G_2} x_i \geq 7,$$

where G_2 is the subset of matching items.

If we also want to restrict the administration time of the test, we can do this by including the following constraint:

$$(9) \qquad \sum_{i=1}^{I} t_i x_i \leq T,$$

where t_i is an estimate of the time an examinee from the population needs for answering item i and T is the upper bound on the administration time for that test.

Another possible kind of constraint reflects the dependencies among items. It is possible that the item bank contains subsets of items V_j ($j = 1, 2, \ldots, J_V$) from which it is not allowed to select more than one item, because every item in such a subset contains information about the answers to the other items in that subset. This demand can be formulated in a linear constraint as follows:

$$(10) \qquad \sum_{i \in V_j} x_i \leq 1, \qquad j = 1, 2, \ldots, J_V.$$

It may also be desirable to select either all or none of the items from a subset W_j ($j = 1, 2, \ldots, J_W$):

$$(11) \qquad \sum_{i \in W_j} x_i = |W_j| x_{ij}, \qquad j = 1, 2, \ldots, J_W,$$

where $|W_j|$ is the number of items in W_j and x_{ij} is an

arbitrary item in W_j.

Sequential Test Construction

If a two-stage test is administered as a paper and pencil test, a decision must be made about the ability levels at which the second tests should aim. Theunissen (1986) shows how these ability levels can be computed, if the tests are constructed sequentially.

If the test is administered by a computer, it is possible to adapt the second test to the individual ability of the examinees, because then the ability of the examinees can be estimated using the score on the routing test. Therefore, we will base the selection of items for the second test on the estimated ability. This implies that the subtests are constructed sequentially.

A test constructor may wish to impose constraints on the item selection at two levels: the subtest or the test level. Both possibilities will be considered in the following two sections.

Constraints at Subtest Level

In this case only constraints at subtest level are considered. A general LP model that selects items for the routing test r can be formulated as follows:

(12) Max. y_r

subject to

$$(13) \qquad \sum_{i=1}^{I} I_i(\theta_l) x_{ir} - y_r \geq 0, \qquad l = 1, \ldots, L,$$

$$(14) \qquad \sum_{i=1}^{I} x_{ir} = n_r,$$

$$(15) \qquad \sum_{i \in K_{rj}} x_{ir} \leq n_{K_{rj}}, \qquad j = 1, \ldots, J_{K_r},$$

$$(16) \qquad \sum_{i \in E_{rj}} x_{ir} = n_{E_{rj}}, \qquad j = 1, \ldots, J_{E_r},$$

$$(17) \qquad \sum_{i \in G_{rj}} x_{ir} \geq n_{G_{rj}} \qquad j = 1, \ldots, J_{G_r},$$

$$(18) \qquad \sum_{i=1}^{I} t_{ir} x_{ir} \leq T_r,$$

$$(19) \qquad \sum_{i \in V_{rj}} x_{ir} \leq 1, \qquad j = 1, \ldots, J_{V_r},$$

$$(20) \qquad \sum_{i \in W_{rj}} x_{ir} = |W_{rj}| x_{ijr}, \qquad j = 1, \ldots, J_{W_r},$$

$$(21) \qquad x_{ir} \in \{0,1\}, \qquad i = 1, \ldots, I,$$

$$(22) \qquad y_r \geq 0.$$

The number of items in the routing test is equal to n_r. In this model the values of p_l are set to one (see the constraints in inequality 2). Because an adaptive test is supposed to measure accurately over the whole ability range, it is assumed that the test constructor wants the same amount of information for each ability level. The advantage of this approach is that the test constructor does not have to specify the relative amount of information that is required at θ_l, $l = 1, \ldots, L$.

In the case of computerized test administration the construction of the second test is based on the estimate θ^* of the ability on the routing test. If the test is administered as a paper and pencil test, then the number of ability levels has to be specified in advance. The second tests are constructed such that they give maximal information on the specified ability levels. A LP model for the construction of a second test s at ability level θ^* is:

$$(23) \qquad \text{Max.} \sum_{i=1}^{I} I_i(\theta^*) x_{is}$$

subject to

$$(24) \qquad \sum_{i \in U} x_{is} = 0,$$

(14) – (17) with subindex r replaced by s,

$$(25) \qquad \sum_{i=1}^{I} t_{is}(\theta^*)x_{is} \leq T_s.$$

(19) – (21) with subindex r replaced by s.

where U is the set of items selected for the routing test. It should be noted that G_{r1} and G_{s1} can be two totally different sets of items. In constraint (25) we suppose that the time needed for answering an item depends on the ability θ^*. In practice, this implies that a large number of response times must be estimated. These estimates are often not available, and it will then not be possible to include constraint (25) in the model. On the other hand, the omission of constraint (25) will enable the use of simple and efficient algorithms for solving the LP model (see Appendix). If simple algorithms cannot be used we have to use heuristics. A heuristic based on the branch–and–bound method that can be used for the present test construction problem is given by Adema (1988).

Constraints at Test Level

If the test constructor specifies demands with respect to the item selection at the level of the test, then the above specified models are no longer applicable. In this section models which can be used for the construction of two–stage tests with demands at test level are given. In these models the constraint on the administration time is omitted because this constraint can rarely be included in view of a lack of data. The LP model for the construction of the routing test r can be formulated as follows:

(26) Max. y_r

subject to

(13) – (14).

(27) $\sum_{i \in E_j} x_{ir} \leq n_{E_j}.$ $j = 1, \ldots, J_E.$

(28) $\sum_{i \in K_j} x_{ir} \leq n_{K_j}.$ $j = 1, \ldots, J_K.$

(29) $\sum_{i \in V_j} x_{ir} \leq 1.$ $j = 1, \ldots, J_V.$

(30) $\sum_{i \in W_j} x_{ir} = |W_j| x_{ijr}.$ $j = 1, \ldots, J_W.$

(21) – (22).

In the model it is supposed that both the number of items in the routing and the second tests are specified separately. The model shows that restrictions on the subsets G_j, $j=1,\ldots,J_G$ are not needed. The way the constraints in (30) are formulated does not allow some of the items of W_j to be in the routing test while the others are in the second test.

The model for the second test is:

(31) Max. $\sum_{i=1}^{I} I_i(\theta^*) x_{is}$

subject to

$$(32) \qquad \sum_{i=1}^{I} x_{is} = n_s.$$

$$(33) \qquad \sum_{i \epsilon U} x_{is} = 0.$$

$$(34) \qquad \sum_{i \epsilon K_j} x_{is} \leq n_{K_j} - \sum_{i \epsilon K_j} x_{ir}. \qquad j = 1, \ldots\ldots, J_K.$$

$$(35) \qquad \sum_{i \epsilon E_j} x_{is} = n_{E_j} - \sum_{i \epsilon E_j} x_{ir}. \qquad j = 1, \ldots\ldots, J_E.$$

$$(36) \qquad \sum_{i \epsilon G_j} x_{is} \geq n_{G_j} - \sum_{i \epsilon G_j} x_{ir}. \qquad j = 1, \ldots\ldots, J_G.$$

$$(37) \qquad \sum_{i \epsilon V_j} x_{is} \leq 1 - \sum_{i \epsilon V_j} x_{ir}. \qquad j = 1, \ldots\ldots, J_V.$$

$$(38) \qquad \sum_{i \epsilon W_j} x_{is} = |W_j| x_{ijs}. \qquad j = 1, \ldots\ldots, J_W.$$

$$(39) \qquad x_{is} \epsilon (0,1), \qquad i = 1, \ldots\ldots, I .$$

where U is the set of items selected for the routing test. In this model the x_{ir}'s are no longer variables, because they are fixed at the value 0 or 1 after the model for the routing test has been solved. The constraints (31) – (33), (38) and (39) can also be found in the LP model for the construction

of second tests with demands at subtest level. These constraints are given again for the sake of clarification.

It is possible that no feasible solution for the second test can be found. This problem can be approached as follows: Let the integer variables z_{E_j} ($j = 1, \ldots, J_E$) and z_{G_j} ($j = 1, \ldots, J_G$) denote the minimum number of items from sets E_j ($j = 1, \ldots, J_E$) and G_j ($j = 1, \ldots, J_G$) to be selected for the second test. Several criteria can be found for partitioning an item bank in subsets of items. Let M denote the number of criteria. The coefficients δ_{mE_j} and δ_{mG_j} are now defined as follows:

$$\delta_{mE_j} = \begin{cases} 0 \text{ if the items are not in } E_j \text{ because of criterion } m \\ 1 \text{ if the items are in } E_j \text{ because of criterion } m \end{cases}$$

$$\delta_{mG_j} = \begin{cases} 0 \text{ if the items are not in } G_j \text{ because of criterion } m \\ 1 \text{ if the items are in } G_j \text{ because of criterion } m. \end{cases}$$

These variables and coefficients are included in the constraints in (27) of the model for the routing test as well as in two new kind of constraints:

$$(40) \qquad \sum_{i \in E_j} x_{ir} + z_{E_j} = n_{E_j}, \qquad j = 1, \ldots, J_E.$$

$$(41) \qquad \sum_{i \in G_j} x_{ir} + z_{G_j} \geq n_{G_j}, \qquad j = 1, \ldots, J_G.$$

$$(42) \qquad \sum_{j=1}^{J_E} \delta_{mE_j} z_{E_j} + \sum_{j=1}^{J_G} \delta_{mG_j} z_{G_j} \leq n_s. \qquad m = 1, \ldots, M.$$

The constraints in (42) exclude the possibility that, for instance, at least 7 addition items and exactly 4 subtraction items must be selected for the second test, while the number of items in the second test is 10.

Simultaneous Test Construction

If the routing and the second test are constructed before they are administered, it is possible to construct the subtests simultaneously instead of sequentially. The choice of the ability levels of the second tests, however, is a new problem. In this paper we will suppose that the ability levels of the second tests can be selected in advance. This can be done safely, because item information functions are continuous, well behaved smooth functions. If practical demands with respect to the item selection are specified at subtest level it is much easier to construct the subtests sequentially. Therefore, only demands at the level of the test are considered. The model that will be given is not always practical, because it includes a large number of decision variables and constraints. We will return to this problem at the end of the section. The advantage of simultaneous test construction compared to sequential test construction is the better distribution of "good" and "bad" items over the routing and second test.

Define the decision variables x_{ip} by:

$$x_{ip} = \begin{cases} 0 & \text{item } i \text{ not in subtest } p \\ 1 & \text{item } i \text{ in subtest } p. \end{cases} \qquad p = 1, \ldots, P$$

The routing test is denoted by $p = 1$. So there are $P - 1$ second tests. In the model y_r is equal to the minimum of

$$\sum_{i=1}^{I} I_i(\theta_l)\, x_{il}$$

for $l = 1, \ldots, L$ and y_s is equal to the minimum of

$$\sum_{i=1}^{I} I_i(\theta_p^*)\, x_{ip},$$

where θ_p^* is the ability level at which second test p is peaked for $p = 2, \ldots, P$. The objective function of the model is to maximize these minima simultaneously. Because the value of y_r respectively y_s is influenced by the number of items in the routing test respectively the second tests, we have to weight the variables y_r and y_s by $1/n_r$ or $1/n_s$ respectively in the objective function. So if we assume that the items in the routing test and the second tests are equally important, then we can formulate the following model:

(43) Max. $y_r/n_r + y_s/n_s$

subject to

$$(44) \qquad \sum_{i=1}^{I} I_i(\theta_1)x_{i1} \geq y_r. \qquad l = 1, \ldots, L.$$

$$(45) \qquad \sum_{i=1}^{I} I_i(\theta_p^*)x_{ip} \geq y_s. \qquad p = 2, \ldots, P.$$

$$(46) \qquad \sum_{i=1}^{I} x_{i1} = n_r.$$

$$(47) \qquad \sum_{i=1}^{I} x_{ip} = n_s. \qquad p = 2, \ldots, P.$$

$$(48) \qquad \sum_{i \epsilon K_j} x_{i1} + \sum_{i \epsilon K_j} x_{ip} \leq n_{K_j}. \qquad j = 1, \ldots, J_K ; \quad p = 2, \ldots, P .$$

$$(49) \qquad \sum_{i \epsilon E_j} x_{i1} + \sum_{i \epsilon E_j} x_{ip} = n_{E_j}. \qquad j = 1, \ldots, J_E ; \quad p = 2, \ldots, P .$$

$$(50) \qquad \sum_{i \epsilon G_j} x_{i1} + \sum_{i \epsilon G_j} x_{ip} \geq n_{G_j}. \qquad j = 1, \ldots, J_G ; \quad p = 2, \ldots, P .$$

$$(51) \qquad \sum_{i \epsilon V_j} x_{i1} + \sum_{i \epsilon V_j} x_{ip} \leq 1. \qquad j = 1, \ldots, J_V ; \quad p = 2, \ldots, P .$$

$$(52) \qquad \sum_{i \in W_j} x_{i1} + \sum_{i \in W_j} x_{ip} = |W_j| x_{i_j 1} + |W_j| x_{i_j p}, \qquad j = 1, \ldots, J_W; \quad p = 2, \ldots, P,$$

$$(53) \qquad x_{i1} + x_{ip} \leq 1, \qquad p = 2, \ldots, P; \quad i = 1, \ldots, I,$$

$$(54) \qquad x_{ip} \in (0,1), \qquad p = 1, \ldots, P; \quad i = 1, \ldots, I,$$

$$(55) \qquad y_r, y_s \geq 0.$$

The above model for simultaneous test construction is not very practical, because the numbers of constraints and variables are so large that the amount of CPU-time and computer memory needed for solving the model may be prohibitive (see Examples). However, it may be possible to make use of the special structure of the model and develop heuristics that are capable of solving the model under realistic conditions.

Multi-stage Testing

Multi-stage testing differs from two-stage testing in that more than one subtest is administered after the routing test. The choice of each subtest depends on the scores on the preceding test. The linear programming models for the

construction of second-stage tests with constraints at subtest level can also be used for the construction of multi-stage tests, provided that one modification is taken into account. In this case, set U in the constraint in (24) must be redefined as the set of items selected for the preceding subtests. The models with constraints specified at test level can also be modified in a straight-forward way, so that they are useful for the construction of multi-stage test procedures. Extension of the model to simultaneous test construction is also possible; but will enlarge the problems associated with large numbers of variables and constraints.

Examples

In this section two examples are given. In the first example a two-stage test is constructed sequentially with the demands specified at test level. In the second example the two-stage test is constructed simultaneously. A simulated item bank for French with 300 items that fitted the 3-parameter model ($a_i \sim U(0.5,1.5)$; $b_i \sim U(-3,3)$; $c_i = 0.2$) was used for both examples. The item bank was partitioned with respect to its content in a vocabulary (items 1-100), a grammar (items 101-200), and a reading comprehension part (items 201 - 300). The first 100 items of those subsets were of the multiple-choice type. The other items were of the matching type.

The demands of the test constructor in both examples were:

- The ability levels at which the target information function is specified for the routing test is $\theta_1 = -1$, $\theta_2 = 0$, and $\theta_3 = 1$.
- The second tests are peaked at the ability levels -1 and 1.
- The number of items in the routing test and second tests is 20.
- The examinee should answer in total less than 15 vocabulary exactly 12 grammar, and more than 15 reading comprehension items.
- The number of multiple-choice items the examinee should answer is smaller than 25. For the matching items this number is greater than 13.

The linear programming models with 0-1 variables that were used in the examples can be solved by a branch-and-bound method (Land & Doig, 1960). The models were solved on a DEC-2060 computer with a modified version of the program Lando (Center for Mathematics and Computer Science). The modifications in the branch-and-bound part of the algorithm are described by Adema (1988). The CPU-times in the examples do not include the time needed for reading the input file, for the initialization and for writing to the output file. The CPU-times are shown to give an impression about the practicality of the approaches.

Example 1

In this example we consider the case of sequential test construction with demands at test level.

An LP model for constructing a routing test that fulfilled the demands was formulated. The routing test constructed with this model was composed as follows: 7 vocabulary items; 4 grammar items; 9 reading comprehension items; 8 multiple-choice items and 12 matching items. The test information value was 4.050 at θ_1, 5.809 at θ_2, and 4.037 at θ_3.

Given the composition of the routing test, the restrictions on the composition of the second tests were:

- The second tests should contain less than 8 vocabulary, exactly 8 grammar, and more than 6 reading comprehension items.
- The second tests should contain less than 17 multiple-choice items and more than 1 matching item.

The information values of the second tests were:

Second test at $\theta^* = -1$: 5.052;

Second test at $\theta^* = 1$: 5.690.

The total CPU-time for constructing the routing test and second tests was 7.3 seconds.

Example 2

In this example the routing test and second tests were constructed simultaneously with LP model (43) - (55). For the constructed routing test the test information value was 3.689 at θ_1, 5.608 at θ_2, and 3.742 at θ_3. The information values of the second tests were:

Second test at $\theta^* = -1$: 5.758;

Second test at $\theta^* = 1$: 5.758.

The CPU-time needed for computing a solution was 659.3 seconds.

Discussion

In this paper linear programming models for the construction of two-stage testing procedures are proposed. The model for simultaneously constructing the subtests has theoretical value as yet, because of the large number of (0,1)-variables and constraints in the model and the problem of specifying the ability levels of the second stage tests. However, it may be possible to develop heuristics for special cases. More promising were the models for sequential test construction. Especially the model with constraints at subtest level is easy to apply, because the routing test and the second test can be constructed separately. When the test constructor specifies the constraints at test level, some problems will arise because we will have to take into account the construction of the second-stage test in the model for the routing test and reverse. These problems can be solved by introducing the constraints in (40) through (42).

There are other forms of adaptive testing which may be more efficient than two-stage testing. The construction of such adaptive tests, using mathematical programming models is not always practical and can be too difficult if not impossible. As an example, the case of tailored testing is considered. In tailored testing one item is selected at a time so that the selection of the best item is easy: Select

the item that gives the most information at the current ability estimate of the examinee and also satisfies the practical constraints. After the selection of an item the practical constraints have to be adjusted, just like we adjusted the constraints in model (31)–(39) to the items selected for the routing test. In fact, model (31)–(39) can be used for the construction of tailored tests by adapting the model in a straight-forward way and choosing $n_s = 1$.

Appendix

In this appendix a simple algorithm is given for solving a model with objective function (23) and constraints (14) through (17) with subindex r replaced by s where the intersection of the subsets is empty. In the first part of the algorithm the items giving the most information at ability level θ^* are selected from the subsets $E_j (j = 1, \ldots, J_E)$ and $G_j (j = 1, \ldots, J_G)$ such that the =–constraints are satisfied and equality yields for the ≥–constraints. In the second part of the algorithm the rest of the n items are selected from the subsets $G_j (j = 1, \ldots, J_G)$ and $K_j (j = 1, \ldots, J_k)$ such that the ≤–constraints are satisfied and the information at ability level θ^* is maximized. For convenience, subscript s is omitted in the following description of the algorithm.

Algorithm:

Step 1: Sort the items in the subsets $E_1, \ldots, E_{J_E}$, $G_1, \ldots, G_{J_G}$ in sequence of decreasing information at ability level θ^*. Go to Step 2.

Step 2: The first n_{E_j} ($j = 1, \ldots, J_E$) and n_{G_j} ($j = 1, \ldots, J_G$) items from the subsets $E_1, \ldots, E_{J_E}$, $G_1, \ldots, G_{J_G}$ are selected for the test. Go to Step 3.

Step 3: Put the items of $K_1, \ldots, K_{J_K}$ and the items of $G_1, \ldots, G_{J_G}$ that are not selected in a queue. Sort the items in the queue in sequence of decreasing information at ability level θ^*. Go to Step 4.

Step 4: If n items are selected the algorithm stops. If less then n items are selected, then take the first item from the queue. Check whether the selection of this item is feasible for the $\leq$ -constraints. If so, add the item to the test. Repeat Step 4.

We can adapt the algorithm so that it can also solve other problems. For instance, it is easy to include constraints (19) with r replaced by s. The purpose of this appendix was just to show that it is sometimes possible to solve a 0-1 LP model by a simple and efficient algorithm.

References

Adema, J.J. (1988). A note on solving large-scale zero-one programming problems. (Research Report 88-4). Enschede: University of Twente, Department of Education.

Fischer, G.H., & Pendl, P. (1980). Individualized testing on the basis of the dichotomous Rasch model. In: L.J.TH. van der Kamp, W.F. Langerak, & D.N.M. de Gruijter (Ed.), Psychometrics for educational debates, John Wiley & Sons.

Glas, C.A.W. (1988). The Rasch model and multistage testing, Journal of Educational Statistics, 13-1, 45-52.

Land, A.H., & Doig, A.G. (1960). An automated method of solving discrete programming problems, Econometrica, 28, 497-520.

Lord, F.M. (1980). Applications of item response theory to practical testing problems, Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Theunissen, T.J.J.M. (1986). Some applications of optimization algorithms in test design and adaptive testing, Applied Psychological Measurement, 10, 381-389.

van der Linden, W.J., & Boekkooi-Timminga, E. (1989). A maximin model for test design with practical constraints, Psychometrika, to appear.

Wainer, H., & Kiely, G.L. (1987). Item clusters and computerized adaptive testing: A case for testlets, Journal of Educational Measurement, 24-3, 158-201.

Yen, W.M. (1983). Use of the three-parameter model in the development of a standardized achievement test. In R.K. Hambleton (Ed.), Applications of item response theory. Vancouver: Educational Research Institute of British Columbia.

Titles of recent Research Reports from the Division of Educational Measurement and Data Analysis, University of Twente, Enschede, The Netherlands.

RR-87-1 R. Engelen, *Semiparametric estimation in the Rasch model*

RR-87-2 W.J. van der Linden (Ed.), *IRT-based test construction*

RR-87-3 R. Engelen, P. Thommassen, & W. Vervaat, *Ignatov's theorem: A new and short proof*

RR-87-4 E. van der Burg, & J. de Leeuw, *Use of the multinomial jackknife and bootstrap in generalized nonlinear canonical correlation analysis*

RR-87-5 H. Kelderman, *Estimating a quasi-loglinear models for the Rasch table if the number of items is large*

RR-87-6 R. Engelen, *A review of different estimation procedures in the Rasch model*

RR-87-7 D.L. Knol & J.M.F. ten Berge, *Least-squares approximation of an improper by a proper correlation matrix using a semi-infinite convex program*

RR-87-8 E. van der Burg & J. de Leeuw, *Nonlinear canonical correlation analysis with k sets of variables*

RR-87-9 W.J. van der Linden, *Applications of decision theory to test-based decision making*

RR-87-10 W.J. van der Linden & E. Boekkooi-Timminga, *A maximin model for test design with practical constraints*

RR-88-1 E. van der Burg & J. de Leeuw, *Nonlinear redundancy analysis*

RR-88-2 W.J. van der Linden & J.J. Adema, *Algorithmic test design using classical item parameters*

RR-88-3 E. Boekkooi-Timminga, *A cluster-based method for test construction*

RR-88-4 J.J. Adema, *A note on solving large-scale zero-one programming problems*

RR-88-5 W.J. van der Linden, *Optimising incomplete sample designs for item response model parameters*

RR-88-6 H.J. Vos, *The use of decision theory in the Minnesota Adaptive Instructional System*

RR-88-7 J.H.A.N. Rikers, *Towards an authoring system for item construction*

RR-88-8 R.J.H. Engelen, W.J. van der Linden, & S.J. Oosterloo, *Item information in the Rasch model*

RR-88-9 W.J. van der Linden & T.J.H.M. Eggen, *The Rasch model as a model for paired comparisons with an individual tie parameter*

RR-88-10 H. Kelderman & G. Macready, *Loglinear-latent-class models for detecting item bias*

RR-88-11 D.L. Knol & M.P.F. Berger, *Empirical comparison between factor analysis and item response models*

RR-88-12 E. van der Burg & G. Dijksterhuis, *Nonlinear canonical correlation analysis of multiway data*

RR-88-13 J. Kogut, *Asymptotic distribution of an IRT person fit index*

RR-88-14 J.J. Adema, *The construction of two-stage tests*

Research Reports can be obtained at costs from Bibliotheek, Department of Education, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.

department of
EDUCATION

A publication by
the Department of Education
of the University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands